



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|-----------------|-------------|----------------------|---------------------|------------------|
| 09/818,214 | 03/26/2001 | Rahul Sharma | SUNMP005 | 2476 |

25920 7590 01/12/2004

MARTINE & PENILLA, LLP
710 LAKEWAY DRIVE
SUITE 170
SUNNYVALE, CA 94085

| |
|----------|
| EXAMINER |
|----------|

LOHN, JOSHUA A

| | |
|----------|--------------|
| ART UNIT | PAPER NUMBER |
|----------|--------------|

2114

DATE MAILED: 01/12/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/818,214

Applicant(s)

SHARMA ET AL.

Examiner

Joshua A Lohn

Art Unit

2114

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 09 October 2003.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-32 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-20, 28, 29 and 32 is/are rejected.
- 7) ☒ Claim(s) 21-27, 30 and 31 is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 26 March 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. §§ 119 and 120

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
* See the attached detailed Office action for a list of the certified copies not received.
- 13) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.
a) ☐ The translation of the foreign language provisional application has been received.
- 14) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892) 4) ☐ Interview Summary (PTO-413) Paper No(s). _____
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948) 5) ☐ Notice of Informal Patent Application (PTO-152)
- 3) ☒ Information Disclosure Statement(s) (PTO-1449) Paper No(s) 4, 5. 6) ☐ Other:

Art Unit: ~~2184~~
2114

FIRST NON-FINAL ACTION

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 1-20, 28, 29, and 32 are rejected under 35 U.S.C. 103(a) as being unpatentable over "Clustering: Transparent Replication, Load Balancing, and Failover" by Salil Deshpande, published January 2000.

As per claim 1, Deshpande discloses executing a Java application on a server, where the Java application includes multiple entity beans, see Figure 1 on page 5, where plural entity beans (EJBs) are executed on the application server. It is obvious that Deshpande discloses a replicated state manager, wherein the replicated state manager includes program instructions for managing a replicated and migration capable session state of the Java application using an in-memory database within a Java server process. This is obvious in the fact that Deshpande provides for replicas of session beans and the ability to transparently fail over to these replicated beans, and any replication and fail over would obviously be controlled by some form of a replicated state manager, see page 14. Deshpande additionally provides for the session state of an entity bean being stored within a database, see page 14, which it would be obvious to provide in-memory, see page 10. It would be obvious to use an in-memory database for replication when using a system that has Servlet session state replication for fail over. Deshpande also obviously provides for the replicated state manager including replicating the in-memory state of the Java application

Art Unit: 2184

to the replicated state server, see page 10, where when using a Servlet session the state is replicated across multiple nodes, this state replication obviously including any entity beans that are part of the session state.

As per claim 2, Deshpande discloses replicating state server data in-memory, see page 10.

As per claim 3, Deshpande teaches that it would be obvious to have a disk-based replication. This would have been obvious because an in-memory replication is more susceptible to failures and not suitable to persistent components.

As per claim 4, Deshpande discloses state replication to different types of state servers, see page 15, where a failure of the currently used state server causes the use of a distributed state server environment that holds the duplicated states.

As per claim 5, it is obvious that the state recovery occurs during application failure, this is obvious because on page 15 Deshpande discloses the fail over occurring during a session that is invoking various entity beans. Any session that is currently accessing entity beans is obviously running an application that is utilizes those entity beans.

As per claim 6, Deshpande discloses a checkpoint policy used to configure the fail over of the entity beans. The checkpoint policy is described as having a correct state provided at the end of each successful transaction, and storing this state in the database, see page 14.

As per claim 7, Deshpande discloses the storing of a state of the entity bean using a state object, where the state object is the correct state stored in the database, see page 14.

As per claim 8, It is described in the rejection of claim 1 above that the states are taught as being able to be stored as in-memory objects, it is obvious that the state is managed with a

Art Unit: 2184

J2EE server process. This is obvious because the use of entity beans, as taught by Deshpande, requires that the J2EE standards be used, see page 4.

As per claim 9, Deshpande teaches of a logical separation between the application and the state objects, see page 15, where the application is accessing entity beans in one container while the state objects that include fail over duplicates exist in a separate logical container.

As per claim 10, Deshpande teaches updating and tracking changes of the state objects, for the checkpoint mechanism, in response to changes in the state of the application. This is shown in the state objects being checkpointed to the state of the last successful transaction, implemented by the application, see page 14.

As per claim 11, Deshpande discloses an application part having an entity bean object, see Figure 1. Deshpande obviously teaches a managed state having an object that stores a state of the entity bean object within a memory address space of a J2EE server process. This is obvious because Deshpande discloses the use of entity beans, as shown above, and this use of entity beans would require the use of a J2EE process on the server which is executing the objects, see page 4. Deshpande also discloses a replicated state server that stores a replica of the state object, where the replicated server is the database that stores the correct state of an entity bean, see page 14.

As per claim 12, Deshpande discloses a memory replicated state server, in the form of in-memory replication that would be obviously used if a Servlet session is being used with fail over, see page 10.

As per claim 13, Deshpande teaches that it would be obvious to have a disk-based replication. This would have been obvious because an in-memory replication is more susceptible to failures and not suitable to persistent components.

As per claim 14, Deshpande teaches of a logical separation between the application and the managed state objects, see page 15, where the application is accessing entity beans in one container while the state objects that include fail over duplicates exist in a separate logical container.

As per claim 15, Deshpande teaches of the replicated state manager tracking updates to the state object in response to changes in the state of the entity bean object as part of a transaction. This is shown in the replicated state being a state of the last successful transaction, which would obviously mean that the replicated state is updated as part of a transaction completing, see page 14.

As per claim 16, Deshpande discloses an application part having a plural entity bean objects, see figure 1. These entity beans are taught to be related, see the example on page 15, where entity beans E1 and E2 are related due to use by the same application. Deshpande also teaches of a managed state having state objects to store the state of each of the entity beans, wherein the relationship of the beans is stored. This is shown in the example on page 15 with the state of the two entity beans being used by the invoking session, where an invoking session would obviously use the state of the entity bean objects. Deshpande also teaches of a replicated state server that stores a replica of the first state object and a replica of the second state object, see the database on page 14 that stores a replica of the state of each entity bean.

As per claim 17, Deshpande discloses a memory replicated state server, in the form of in-memory replication that would be obviously used if a Servlet session is being used with fail over, see page 10.

As per claim 18, Deshpande teaches that it would be obvious to have a disk-based replication. This would have been obvious because an in-memory replication is more susceptible to failures and not suitable to persistent components.

As per claim 19, Deshpande teaches of a logical separation between the application and the managed state objects, see page 15, where the application is accessing entity beans in one container while the state objects that include fail over duplicates exist in a separate logical container.

As per claim 20, Deshpande teaches of the replicated state manager tracking updates to the state object in response to changes in the state of the entity bean object as part of a transaction. This is shown in the replicated state being a state of the last successful transaction, which would obviously mean that the replicated state is updated as part of a transaction completing, see page 14.

As per claim 28, Deshpande discloses a checkpoint policy used to configure the fail over of the entity beans. The checkpoint policy is described as being configured to have a correct state provided at the end of each successful transaction, and storing this state in the database, see page 14.

As per claim 29, Deshpande discloses a synchronous checkpoint method in which the checkpoints are stored as they are generated in response to the successful completion of a transaction, see page 14.

Art Unit: 2184

As per claim 32, Deshpande discloses the checkpoints being made in response to single transactions, see page 14.

Allowable Subject Matter

As per claim 21-27, 30, and 31, these claims are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure is provided on form PTO-892.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Joshua A Lohn whose telephone number is (703) 305-3188. The examiner can normally be reached on M-F 8-4.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Robert Beausoleil can be reached on (703) 305-9713. The fax phone number for the organization where this application or proceeding is assigned is (703) 746-7239.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

JAL


SCOTT BADERMAN
PRIMARY EXAMINER